Universal Serial Bus (USB)
How Many Ports
and
How To Get There?

R1.0
6/9/99

An Intel white paper outlining how many ports an Easy PC should have for Back To School (BTS) 1999 and how to implement them

Author: Alex Warshofsky
For: DPG-M

Content Providers and Reviewers:
Matt Dunford
Robert Hale
Brad Hosler
John Hyde
John Lusk
Mark Skarpness
Stephen Whalley
Many others.

Table of Contents

This page intentionally left blank

1 – Introduction

On average, there are over 10 different cable types that connect to the back of a PC (for example, PS/2, serial, parallel, microphone in, line in, line out, power in, VGA out, in/out telephone jacks, Ethernet, and SCSI). Before first use of a computer, someone has to connect each cable to its corresponding port.

One of the many uses of Universal Serial Bus (USB) is to reduce the number of different cable types going into the computer, improving setup time, and making the computer easier to use.

2 – Background

Definition

Universal Serial Bus (USB) is best described in the USB Specification found at the *www.usb.org/developers/download.htm* website with the following quote:

> "The USB is a cable bus that supports data exchange between a host computer and a wide range of simultaneously accessible peripherals. The attached peripherals share USB bandwidth through a host-scheduled, token-based protocol. The bus allows peripherals to be attached, configured, used, and detached while the host and other peripherals are in operation."

In everyday day language, USB is a way of connecting peripherals to a host, usually a PC.

Why USB? The need for a different approach

In 1981, when the first PC was introduced, there were relatively few ports in the rear of the box. As time passed, the complexity of the computer increased with more ports and new connector types. For the home, audio ports were added. In the office, networking became ubiquitous. (In fact, three different kinds of network connectors were used.) External drives proliferated. Now cameras are a common add-on peripheral. The end result is a PC looking like a very scary octopus, since every connector tentacle is different.

So what is the big deal with having all these cables? It is not just that it looks bad and makes the desk cluttered. A much larger problem is that the computer is harder to use. Many first-time buyers are afraid of computers and part of that comes from the setup process. Making the installation and setup process easier results in higher customer satisfaction. This may reduce the computer's "hard-to-use" aura and may increase total sales in the industry.

Upgrading an existing system is another area that can be facilitated with the use of the USB port. If an end user buys a peripheral, such as a scanner, and wants to plug it into their system he/she must find the correct port to plug it into before the scanner can be useful. This may take a while, especially if this person is a novice user and has to read the manual to figure out where to connect the cable. In an extreme case, the end user could stare at the back of their computer trying to find where the correct port is without realizing that some other device already uses it. Or, the computer could be neatly stored under the computer desk where someone has to get on their hands and knees to inspect the rear port connections. Or worse yet, what if the computer has to be opened and a proprietary I/O card for the scanner has to be installed. Some users may say it is not worth it and return the scanner.

If a USB hub was built into the monitor or front panel of the computer, all the end user needs to do is plug the scanner, with the provided USB cable, into a device that may already be on the desktop. Monitors are always within a hand's reach. Again, USB makes it easier to upgrade and this will result in higher customer satisfaction.

3 – USB Feature Set

USB was developed with many features in mind. The following presents a list of features that make USB so compelling:

Bandwidth — USB has a maximum throughput of 12 Mbps with an option of a low cost 1.5 Mbps interface. In comparison, the common serial port has a bandwidth of 110 Kbps and the parallel port has 2 Mbps. This high bandwidth has two advantages. One is that high bandwidth devices such as cameras can be used. The other advantage is that multiple devices can share this bandwidth and co-exist on the same controller.

Expandable — One USB port can turn into 7 connectable ports by using a hub, which can then be turned into 49. (See description below for more on hubs.) This can be done using multiple tiers of hubs up to a depth of 5. As a side note, most USB hubs only implement a fan-out of 4.

Hot Swappable — Under some operating systems (for example, Microsoft* Windows* 98), a USB device can be plugged into a USB port at any time and then be instantly recognized and configured by the operating system (OS). On non plug-n-play OS's, a driver will still need to be loaded. (Fortunately, over 90% of all PCs use a plug-n-play OS.) A USB connected device can also be unplugged at any time with no harm to the system. This is very different from a SCSI device, which needs to be connected to the PC and powered before the computer boots up if the computer is to recognize it without user intervention. Otherwise the user must go through a series of steps that are quite cumbersome to get the OS to recognize the peripheral.

Cost — The total cost of adopting USB is relatively low. The actual cable is similar to a phone cable. The USB cable has 4 wires with some shielding for the high-speed cable and no shielding for the low speed wire. (Compare this to SCSI, which has a minimum of 25 wires) Since the bus protocol can be implemented using a minimal number of gates (1500), all the control functions can be done in one ASIC. This ASIC could easily be integrated into the logic that is already on most peripherals. (A synthesizable core can be purchased from multiple vendors, many of whom are listed at the *www.usb.org* website. Although the USB silicon is more expensive than the traditional serial port[1], the total additional cost of implementing USB is minimal and can benefit from economy of scale as volume sales increase. Retail price parity has already been achieved with some devices; for example, a Microsoft* PS2 mouse and a Microsoft* USB mouse are selling for the same price at many retail outlets.

Universal — The technology is general enough that almost all peripherals can be switched to USB, including speakers, microphones, secondary storage devices, and network devices (analog modems, DSL, and Ethernet).

Bus Powered — The bus itself provides up to 500 mA to each peripheral. This means that low power devices such as video conferencing equipment will not require a separate power supply; thereby reducing the total number of cables around the system.

[1] The traditional serial port is not bus based and therefore requires almost no logic. USB is also a serial port, but it is an intelligent port.

4 – Uses of USB

USB can be used as the interface to over 90% of all available peripherals on the market today ranging from simple mice, keyboards, speakers and microphones, all the way to the complex CD-rewritables and Ethernet adapters. See Appendix B for a brief list of currently available peripherals.

The number of USB devices in the market has been increasing over the last few years. This coupled with PC manufacturers moving away from legacy support, will result in an increase of the number of USB devices attached to the PC. As this number increases two problems may occur. The first one is a lack of ports, and the second problem is a lack of bandwidth.

Number of Ports — It would be very convenient if more than two USB devices could plug directly into the PC. Every PC requires a mouse and a keyboard, which requires 2 USB ports. In the age of the Internet, a common PC would include additional USB ports for a camera and microphone for video conferencing. Needless to say, two ports would not suffice for this application. This does not even take into account the networking device that would be required to connect to the outside world.

Bandwidth — Port quantity is not the only problem. In the common set up described above, port throughput is also an issue. Table 1 lists some common devices and their throughputs. The total required throughput for the above example is approximately 7 Mbps from 6 different devices. 7 Mbps is sufficiently under the 12 Mbps maximum bandwidth of USB that no problems should occur. However, should the bandwidth on either the camera (higher frame rate or less compression in camera) or the modem (switch to ADSL*, 8 Mbps, instead of G.Lite*, 1.5 Mbps) increase, the USB bus will become saturated and the OS will cause the bulk transfers (most likely the modem) to decrease.

Table 1[2]

Device	**Bandwidth**	**Device**	**Bandwidth**
Keyboard	100 bps	Speakers	700 Kbps
Mouse	<100 bps	Video Camera	4 Mbps
Microphone	350 Kbps	Ethernet	10 Mbps
Analog Modem	56K bps	G. Lite DSL	1.5 Mbps
Cable Modem	2Mbps	ADSL	8 Mbps

[2] See Appendix A for calculation details.

5 – Immediate Ways of Adding USB

In 1996, Intel started to include a USB controller in its chipsets. Now, very few systems are shipped without some type of USB support. The following section describes a few of the easier ways of adding USB ports to a computer. Some of the options can be used with each other to add multiple ports.

Internal

Intel Chipsets — Just about every motherboard being produced has two USB ports built onto them. This is done using the host controller that is built into the chipset. The only added parts that are required are the physical ports, power regulator and a few discrete passive devices. The total cost of these parts is approximately $1. Reference designs are easily obtained by contacting the appropriate people. (for example, at the *http://developer.intel.com/* site).

Add-in Cards for PCI — This is by far the easiest way of adding USB support to a PC. These come in a few flavors, mainly varying the number of ports available to the end user. The manufacturer's suggested retail price (MSRP) ranges from $30 on up. However, in volume directly from a manufacturer, the cost to an original equipment manufacturer (OEM) could be less than $10. Obviously this is a relatively expensive route, but requires no design work and is easy to assemble. (A list of PCI add-in card vendors can be found at the *www.usb.org* site.)

External

A hub is a way of expanding the number of ports, assuming the computer already has a built-in controller. A hub can be plugged into any USB port. The hub will have at least 2 ports on it and may have up to 7 ports (more is possible, but it is not common). One concept to remember when dealing with hubs is that the bandwidth does not increase. So everything down the line from the controller shares the same 12 Mbps throughput. This could be a problem if multiple high bandwidth devices want to transmit at the same time.

There are two types of hubs: *bus powered* and *self powered*. USB devices are capable of drawing 500 mA of power from each port, if the hub is self powered (meaning it has a power source of its own). If the hub is bus powered, it can only draw 500 mA from its source, so all the devices below it on the chain and the hub itself can only draw a total of 500 mA instead of 500 mA each. Most hubs implement this by allowing each port to draw 100mA each.

Discrete External Hubs — A very simple solution is to add an external discrete hub with its own power supply. This will increase the number of ports by a minimum of 2, and usually by 4. Unfortunately, this is a fairly expensive approach. Since the sole purpose of this device is to act as a hub, there can be no co-mingling of functions (leveraging of costs). Much of the expense of an external device is in the case and the power supply.

Devices with Built-in Hub — Examples of these could be the monitor or keyboard. Merging functionality into one device can save money. In this case, the control logic could be merged into the logic needed for the peripheral. In the case of the monitor, the power supply is already built in and there is no cost for building a case. The advantage of these types of hubs over discrete hubs is that the keyboard and monitor are already on the desktop. Consumers are already familiar with keyboards and adding a hub does not make them any bigger. So not only is the total cost cheaper, but it is more user friendly.

The down-side of using a keyboard as a hub is that it is not powered. This will prevent any device down the chain from drawing the full 500 mA. Cameras and other power hungry devices can not be downstream from a bus-powered hub, unless the device has its own power supply. Mice and gaming devices are ideal to put on this type of hub.

6 – Advanced Methods for Adding USB Ports/Controllers

Integrated Hub on Motherboard

If a computer manufacturer wants to add extra USB ports to the computer box without adding another controller, a hub could be added to the motherboard. This will be cheaper than adding a controller, since a hub is less complex than a controller (fewer gates on the ASIC and fewer pins). Also the design on the motherboard will be easier since USB is only 4 wires and PCI is well over 30 wires, so the layout would be easier (remember that usually easier = cheaper). Hub ASICs are available for slightly over one dollar. The total cost of having 4 ports would be slightly over three dollars[3]. Remember, that this only adds ports and not bandwidth.

Integrated PCI to USB Adapter

A discrete USB host controller could be added directly to the PCI bus. A single ASIC design is more than capable of doing the translation. The cost of the ASIC is estimated to be $3.50 or higher. Of course, the ASIC is not the only added cost. There are a few discrete parts along with the connectors themselves. The approximate cost is $5.50 ($3.50 ASIC + $0.75ports +$1.25misc). The advantage of this type of design is that it not only has the number of ports increased, but total throughput has increased as well. This is a key feature considering the number of high bandwidth devices being used. The following site from Lucent has an example of how to build a PCI to USB adapter card. To transform that design to a motherboard design would require minimal engineering time. (See the website at *http://www.lucent.com/micro/suite/usbprodoc.html*).

Integrated in Super IO

Integrated on motherboard through Super IO (SIO). A possibility is to add a USB controller onto one of the other buses on the motherboard. Since ISA is being removed, the only choice left is the Low Pin Count (LPC) bus. It would be ideal to integrate the Super IO ASIC, which sits on the LPC bus, with a USB controller. Unfortunately, this option has been ruled out due to bandwidth limitations on the LPC bus.

[3] The cost of materials for having just two USB ports is approximately $1. To attain four ports (the USB port from the ICH goes to the hub, which has four ports), the cost of materials would be: hub + 2*2 ports + discrete parts = slightly above 3 dollars. A 5 port solution might be created for less than a $4.00 cost.

7 – Technological Issues

OS support

Microsoft Windows* 98SE, Microsoft Windows* 98, Microsoft Windows* 95 OSR2.1, Linux*, and the BeOS* have USB support.

Hardware Support

USB host controllers on PCI add-in cards may not support legacy keyboards and mice. The reason for this is that certain signals are not routed to the PCI bus, such as the System Management Interrupt (SMI). Whatever is controlling the keyboard or mouse needs to be able to toggle the SMI line. Whenever the keyboard/mouse controller detects any user input, it toggles SMI. When this occurs, the processor goes out and reads the data from the controller. As an add-in card, this is not possible, however, a controller on the motherboard would have access to this signal, so a controller on board could give full support of legacy devices, assuming the BIOS supports the controller.

The need for SMI support is only required prior to OS boot and depends on how the BIOS implements legacy keyboard support. After the OS boots, assuming it has USB support, it no longer requires legacy support and can use the USB keyboard or mouse directly.

A point that must be remembered, is that each USB port requires 500 mA of current at 5 V (2.5 W max) when the system is on and 100mA when the system is in standby. The system vendor must realize this and make sure the power supply is adequate. Please read the USB specification for further details.

BIOS Support

There are two different implementations of USB host controllers based on the USB specification. Most discrete PCI host controllers are implemented with Open Host Controller Interface (OHCI), while the chipsets are usually implemented with Universal Host Controller Interface (UHCI). From the device point of view, these are exactly the same, but the software interface is slightly different, therefore the BIOS needs to be written differently.

Many of the recent BIOS support both types of HCI's; however, they will not support legacy use on both of the HCI's. On Intel motherboards, as of now, legacy support is only being supported on UHCI from the chipset. If a USB keyboard is plugged into an OHCI controller on the PCI bus, the BIOS will not find it on boot up and will give an error along the lines of "Keyboard not found." (This is due to both BIOS issues and hardware issues described above.)

This can be solved in a few different ways:

1. Include legacy support for both HCI's. This has a few major drawbacks. The first problem is that the size of the BIOS must increase to support both HCI's. The expected increase is approximately 4KB of code and another 7 KB for the data space. Another drawback is the time it would take to write the code. The current estimate is that it would take 4 months to write and validate the code. (Please contact a BIOS vendor for exact space and time estimates for any particular board.)
2. Require the end user to only plug the keyboard into the UHCI ports. This could be accomplished by color coding the keyboard cable as it is done now with the PS/2 ports. *However this not very user friendly and is not recommended.* This would cause errors if the user ever switched to a hub off the monitor, which might not be attached to a UHCI controller.

8 – Conclusion

- It is a requirement to have at least 4 ports on any system that expects USB to be the common bus between the PC and the peripherals. Due to current BIOS support, the best way of doing this is with a hub, not a controller. (Having USB ports that behave differently is unacceptable.) The best place to add the hub will be on the motherboard or the monitor. This way, no extra power supplies will be required.
- As time goes by, it is very likely that the other alternatives listed in this document will become viable. Once the BIOS issues are resolved, switching to a second controller on the motherboard would be preferred instead of using a hub. As stated earlier in this document, the added controller would increase the available bandwidth from the PC.
- The following table describes the options available to increase the number of ports. It includes both the effectiveness and the cost of each option. These choices have been ranked from the "best" choice to the least ideal option. (Best being defined as having the highest functionality and the least cost.)

	Added cost to OEM	**Increase throughput**	**Total number of ports**	**Motherboard redesign**	**Time to market**	**Full Function-ality**
Hub on Motherboard	$4	NO	4	YES	MED	Yes
Discrete External Hub	$9	NO	4	NO	FAST	Yes
Internal PCI→USB	$7	YES	4	YES	SLOW	NO
PCI add-in card	$10	YES	4	NO	FAST	NO

Notes:

1. All costs are estimated cost to manufacture.
2. Total number of ports may vary depending on actual implementation.
3. Time to market reflects the author's opinion of when an OEM could have a system on the market. FAST suggests immediate, MED is by Fall of 1999, and SLOW would depend on the amount of resources going into it.
4. Functionality is for the current state. As stated early in this paper, legacy keyboard support is not supported in the BIOS for any controller on the PCI bus other than the chipset itself. This could change with significant engineering effort.

Appendix A – Calculations

Calculations used for determining bandwidth

Microphone = 1 channel * 44 KHz * 8 bps = 350 Kbps

Speakers = 2 channels *44 KHz sampling * 8 bits per sample = 700 Kbps

Video Camera = 352 X 288 (resolution) * 8 bits/pixel * 15 frames a second = 12 Mbps. Modern camera's now do compression at the source, so the required bandwidth is closer to 4 Mbps.

Common setup = Keyboard (100bps) + Mouse (100bps) + Speakers (700Kbps) + Video Camera (4Mbps) + Microphone (350Kbps) + G.Lite modem(1.5Mbps) = 6.5Mbps

Appendix B – Currently Available Peripherals

Legacy
- Mice
- Keyboard

Printers
Speakers
Microphones
Mass Storage
- HD
- Backup
- CD-RW

Cameras
Scanners
Communications
- Analog Modems
- DSL
- Cable modem
- Ethernet